

BAB II

TINJAUAN PUSTAKA

II.1 Pengertian Sistem

Menurut Kusriani, dan Andri Koniyo (2007:5) sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Dengan demikian di dalam suatu sistem, komponen-komponen ini tidak dapat berdiri sendiri-sendiri, tetapi sebaliknya, saling berhubungan hingga membentuk satu kesatuan sehingga tujuan sistem itu dapat tercapai.

Secara garis besar ada 2 (dua) kelompok pendekatan sistem, yaitu :

1. Pendekatan sistem yang lebih menekankan pada elemen-elemen atau kelompoknya, yang dalam hal ini sistem itu didefinisikan sebagai “suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu aturan tertentu.
2. Pendekatan sistem sebagai jaringan kerja dari prosedur, yang lebih menekankan urutan operasi di dalam sistem.

II.1.1. Karakteristik Sistem

Adapun karakteristik yang dimaksud adalah sebagai berikut :

a. **Komponen Sistem (*Components*)**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang saling bekerja sama membentuk suatu komponen sistem atau bagian-bagian dari sistem.

b. Batasan Sistem (*Boundary*)

Merupakan daerah yang membatasi suatu sistem dengan sistem yang lain atau dengan lingkungan kerjanya

c. Subsistem

Bagian-bagian dari sistem yang beraktivitas dan berinteraksi satu sama lain untuk mencapai tujuan dengan sasarannya masing-masing.

d. Lingkungan Luar Sistem (*Environment*)

Suatu sistem yang ada di luar dari batas system yang dipengaruhi oleh operasi sistem.

e. Penghubung Sistem (*Interface*)

Media penghubung antara suatu subsistem dengan subsistem lain. Adanya penghubung ini memungkinkan sebagai sumber daya mengalir dari subsistem ke subsistem lainnya.

f. Masukan Sistem (*Input*)

Energi yang masuk kedalam system, berupa perawatan dari sinyal. Masukan perawatan adalah energy yang dimasukkan supaya sitem tersebut dapat berinteraksi.

g. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

h. Pengolahan Sistem (*Proses*)

Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan mengubah masukan menjadi keluaran.

i. Sasaran Sistem (*Object*)

Tujuan yang ingin dicapai oleh system, akan dikatakan berhasil apabila mengenai sasaran atau tujuan.

II.2. Pengertian Informasi

Menurut Kusrini, dan Andri Koniyo (2007:7-8) informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sumber informasi. Data belum memiliki nilai sedangkan informasi sudah memiliki nilai. Informasi dikatakan bernilai bila manfaatnya lebih besar disbanding biaya untuk mendapatkannya.

II.2.1 Kualitas Informasi

Informasi yang berkualitas memiliki 3 kriteria :

a. Akurat (*Accurate*)

Informasi harus bebas dari kesalahan, tidak bias ataupun menyesatkan. Akurat juga berarti bahwa informasi itu harus dapat dengan jelas mencerminkan maksudnya.

b. Tepat pada waktunya (*Timelines*)

Informasi yang datang pada penerima tidak boleh terlambat. Di dalam pengambilan keputusan, informasi yang sudah usang tidak lagi bernilai. Bila informasi datang terlambat sehingga pengambilan keputusan terlambat dilakukan, hal itu dapat berakibat fatal bagi perusahaan.

c. Relevan (*Relevance*)

Informasi yang disampaikan harus mempunyai keterkaitan dengan masalah yang akan dibahas dengan informasi tersebut. Informasi harus bermanfaat bagi pemakainya. Di samping karakteristik, nilai informasi juga menentukan kualitasnya. Nilai informasi (*value of information*) ditentukan oleh dua hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih besar disbanding biaya untuk mendapatkannya.

II.3. Pengertian Sistem Informasi

Menurut Kusriani, dan Andri Koniyo (2007:8-9) Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.

Dalam suatu sistem informasi terdapat komponen-komponen sebagai berikut :

- a. Perangkat keras (*hardware*), mencakup berbagai peranti fisik seperti komputer dan printer.
- b. Perangkat lunak (*software*) atau program, yaitu sekumpulan instruksi yang memungkinkan perangkat keras memproses data.
- c. Prosedure, yaitu sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.

- d. Orang, yaitu semua pihak yang bertanggung jawab dalam pengembangan system informasi, pemrosesan dan penggunaan keluaran sistem informasi.
- e. Basis data (*database*), yaitu sekumpulan tabel, hubungan dan lain-lain yang berkaitan dengan penyimpanan data.
- f. Jaringan komputer dan komunikasi data, yaitu sistem penghubung yang memungkinkan sumber (*resources*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

II.4 Sistem Informasi Akuntansi

Menurut George H. Bodnar dan William S. Hopwood (2006 : 3) Sistem Informasi Akuntansi (SIA) merupakan kumpulan sumber daya, seperti manusia dan peralatan, yang dirancang untuk mengubah data keuangan dan data lainnya kedalam informasi. Informasi tersebut dikomunikasikan kepada para pembuat keputusan. Sistem informasi akuntansi melakukan hal tersebut entah dengan sistem manual atau melalui sistem terkomputerisasi.

II.5 Retur Penjualan

Menurut Al.Haryono Jusup (2005:327-328) dalam bukunya Dasar-Dasar Informasi Akuntansi, memberikan pengertian bahwa: Retur penjualan adalah pembatalan atas penjualan yang telah dilakukan perusahaan (baik sebagian aatau seluruhnya). Pengembalian barang bisa dilakukan oleh konsumen karena barang tersebut cacat, atau rusak dalam pengiriman, atau sebab lain sehingga pembeli tidak puas.

II.6 Konsep UML (Unified Modelling Language)

Menurut Munawar (2005:17-23) *Unified Modelling Language (UML)* adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembangan sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk

Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan class dan operation dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C. Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan syntax/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML syntax mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: Grady Booch OOD (Object-Oriented Design), Jim Rumbaugh OMT (Object Modeling Technique), dan Ivar Jacobson OOSE (*Object-Oriented Software Engineering*).

Abstraksi konsep dasar UML yang terdiri dari structural classification, dynamic behavior, dan model management, bisa kita pahami dengan mudah

apabila kita melihat gambar diatas dari Diagrams. Main concepts bisa kita pandang sebagai term yang akan muncul pada saat kita membuat diagram. Dan view adalah kategori dari diagram tersebut. Untuk menguasai UML, sebenarnya cukup dua hal yang harus kita perhatikan:

1. Menguasai pembuatan diagram UML
2. Menguasai langkah-langkah dalam analisa dan pengembangan dengan UML.

Sejarah UML sendiri cukup panjang. Sampai era tahun 1990 seperti kita ketahui puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia. Diantaranya adalah: metodologi booch, metodologi coad, metodologi OOSE, metodologi OMT, metodologi shlaer-mellor, metodologi wirfs-brock, dsb. Masa itu terkenal dengan masa perang metodologi (method war) dalam pendesainan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri, yang mengakibatkan timbul masalah baru apabila kita bekerjasama dengan group/perusahaan lain yang menggunakan metodologi yang berlainan.

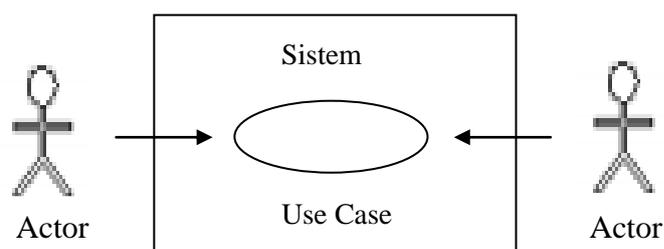
Di sisi lain, ada alat bantu lainnya yang sangat sering dipakai oleh sistem analisa dan perancang. Alat bantu tersebut adalah diagram. Adapun yang termasuk diagram – diagram pada UML yaitu :

1. Use Case diagram.
2. Activity diagram.
3. Class diagram.
4. Sequence diagram.

II.6.1 Diagram – diagram Pada Metode UML

1. Use Case Diagram

Use case adalah alat bantu terbaik guna menstimulasikan pengguna potensial untuk mengatakan tentang suatu sistem dari sudut pandangnya. Tidak selalu mudah bagi pengguna untuk menyatakan bagaimana mereka bermaksud menggunakan sebuah sistem. Karena sistem pengembangan tradisional sering ceroboh dalam melakukan analisis, akibatnya pengguna seringkali susah menjawabnya tatkala dimintai masukan tentang sesuatu. Ide dasarnya adalah bagaimana melibatkan penggunaan sistem di fase – fase awal analisis dan perancangan sistem. Diagram use case menunjukkan 3 aspek dari sistem yaitu actor, use case dan sistem / sub sistem boundary. Actor mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan use case. Gambar II.1 mengilustrasikan actor, use case dan boundary.

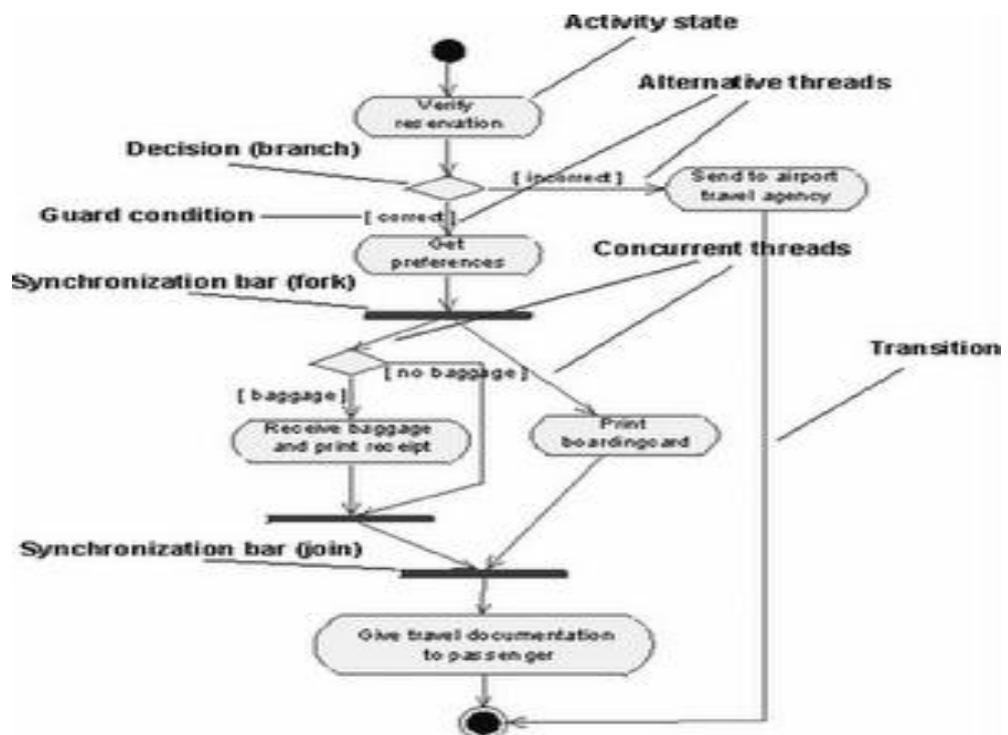


Gambar II.1. Use Case Model

Sumber : Munawar (2005)

2. Activity Diagram

Activity diagram adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. Activity diagram mempunyai peran seperti halnya flowchart, akan tetapi perbedaannya dengan flowchart adalah activity diagram bias mendukung perilaku paralel sedangkan flowchart tidak bisa. Berikut gambar dari sederhana dari Activity diagram.

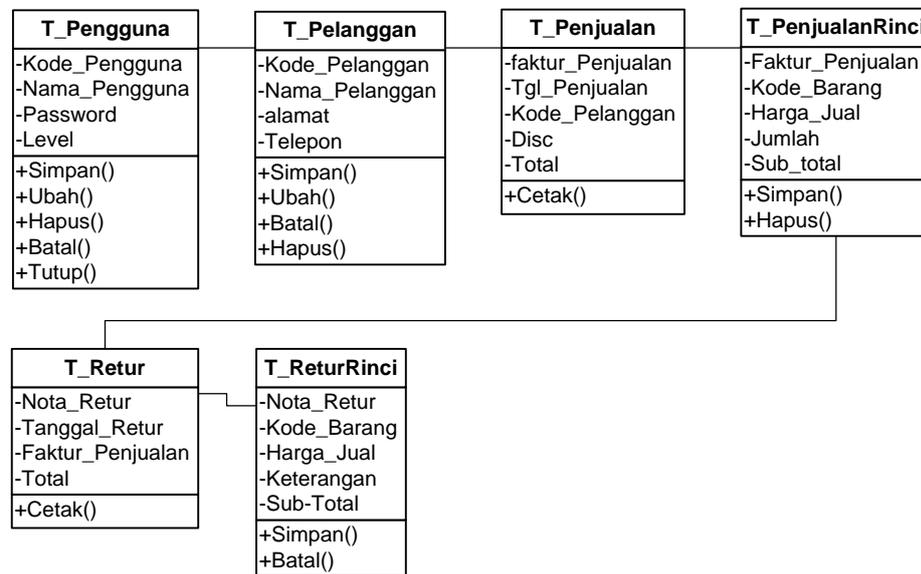


Gambar II.2. Contoh Activity Diagram

3. Class Diagram

Class diagram sangat membantu dalam visualisasi struktur kelas dalam suatu sistem. Hal ini disebabkan karena class adalah deskripsi kelompok

objek – objek dengan property, perilaku (operasi) dan relasi yang sama. Disamping itu class diagram bisa memberikan pandangan global atas sebuah sistem. Hal tersebut tercermin dari class-class yang ada dan relasinya satu dengan yang lainnya. Itulah sebabnya class diagram menjadi diagram paling populer di UML. Contoh gambar class diagram adalah sebagai berikut :



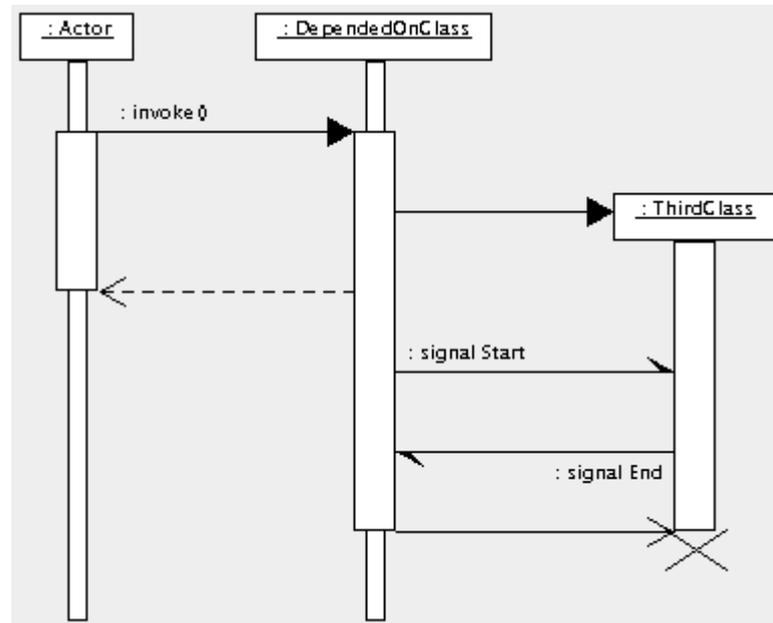
Gambar II.3. Contoh Class Diagram

4. Sequence Diagram

Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah scenario. Diagram ini menunjukkan sejumlah contoh obyek dan pesan yang diletakan diantara obyek – obyek ini di dalam use case.

Komponen utama sequence diagram terdiri atas obyek yang dituliskan dengan kotak segiempat bernama. Message diwakili oleh garis dengan

tanda panah dan waktu yang ditunjukkan dengan progress vertical. Berikut gambar sequence diagram



Gambar II.4. Contoh Sequence Diagram

II.7 Entity Relationship Diagram – ERD

II.7.1. Model-model Data

Menurut Janner Simarmata dan Imam Prayudi (2006 : 59-60) Struktur yang mendasari suatu basisdata adalah model data yang merupakan kumpulan alat-alat konseptual untuk mendeskripsikan data, relasi data, data semantik, dan batasan konsistensi. Untuk mengilustrasikan konsep model data, berikut disajikan dua model data, yaitu *entity relationship model* dan *relational model*. Kedua model menyediakan cara mendeskripsikan rancangan basisdata pada tingkatan logis.

II.7.2. Entity Relationship Model

Entity Relationship (ER) data model didasarkan pada persepsi terhadap dunia nyata yang tersusun atas kumpulan objek-objek dasar yang disebut entitas dan hubungan antar objek. Entitas adalah sesuatu atau objek dalam dunia nyata yang dapat dibedakan dari objek lain. Sebagai contoh, masing-masing mahasiswa adalah entitas dan mata kuliah dapat pula dianggap sebagai entitas.

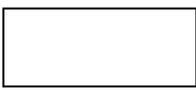
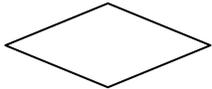
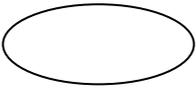
Entitas digambarkan dalam basisdata dengan kumpula atribut. Misalnya atribut nim, nama, alamat dan kota bisa menggambarkan data mahasiswa tertentu dalam suatu universitas. Atribut-atribut membentuk entitas mahasiswa. Demikian pula, atribut kodeMK, namaMK, dan SKS mendeskripsikan entitas mata kuliah.

Atribut NIM digunakan untuk mengidentifikasi mahasiswa secara unik karena dimungkinkan terhadap dua mahasiswa dengan nama, alamat, dan kota yang sama. Pengenal unik harus diberikan pada masing-masing mahasiswa.

Relasi adalah hubungan antara beberapa entitas. Sebagai contoh, relasi menghubungkan mahasiswa dengan mata kuliah yang di ambilnya. Kumpulan semua entitas bertipe sama disebut kumpulan entitas (*entity set*), sedangkan kumpulan semua relasi bertipe sama disebut kumpulan relasi (*relationship set*).

Struktur logis (skema *database*) dapat ditunjukkan secara grafis dengan diagram ER yang dibentuk dari komponen-komponen berikut :

Tabel II.1. Notasi ERD (*Entity Relationship Diagram*)

Notasi	Keterangan
	Entitas , adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
	Relasi , menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
	Atribut , berfungsi mendeskripsikan karakter entitas (atribut yang berfungsi sebagai key diberi garis bawah)
	Garis , sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

Sumber : Janner Simarmata dan Imam Prayudi (2006 : 59-60)

II.8 Microsoft Visual Basic.NET

Menurut wahana komputer (1-2) Visual Basic.NET merupakan suatu lingkungan (environment) terintegrasi untuk membangun dan melakukan uji coba (testing and debugging) berbagai macam aplikasi. Di antaranya adalah aplikasi-aplikasi Windows, aplikasi-aplikasi web, control, class, serta aplikasi-aplikasi console. Visual Basic.NET banyak sekali membuat perubahan-perubahan disbanding dengan visual Basic sebelumnya. Dengan Visual Basic.NET, anda akan lebih mudah membuat aplikasi karena dalam Visual Basic.NET ada dukungan fasilitas baru yang ditambahkan, antara lain Integrated Development Environment (IDE), debugging yang lebih baik dan kemampuan dalam XML Web Service.

Pada dasarnya Visual Studio.NET didesain untuk menampung berbagai macam bahasa pemrograman dan terlingkup dalam Visual Basic.NET. Dengan demikian, Anda dapat membangun aplikasi-aplikasi Windows di dalam Visual Basic.NET. Visual Basic.NET merupakan salah satu bahasa pemrograman yang dapat digunakan untuk membuat program aplikasi. Program ini sangat mudah digunakan dan merupakan bahasa terpopuler untuk saat ini. Bahasa pemrograman ini menyediakan beberapa tool untuk otomatisasi proses pengembangan, yaitu visual tool yang digunakan untuk melakukan beberapa operasi pemrograman dan desain umum, dan juga fasilitas-fasilitas lain yang dapat menunjang dalam pemrograman.

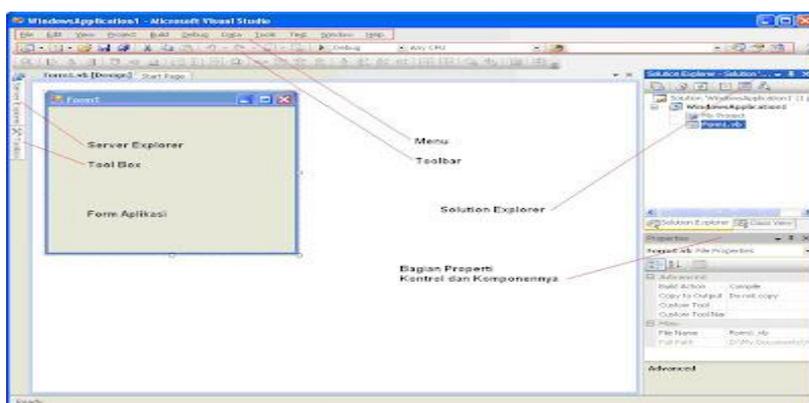
II.8.1. Mengenal Layar Kerja *Visual Basic.Net 2008*

Saat menjalankan *Visual Basic.Net* untuk pertama sekali akan muncul kotak dialog, seperti yang terlihat pada Gambar II.5.



Gambar II.5 : Jendela Dialog Awal *Visual Basic.Net 2008*

Pada kotak dialog ini kita bisa memilih jenis aplikasi apa yang akan kita buat dengan *Visual Basic*. Setelah *Visual Basic* dijalankan, akan muncul jendela kerja yang ditunjukkan seperti Gambar II.6.



Gambar II.6 : Lembar kerja *Visual Basic.Net 2008*

Keterangan gambar :

a. Menu

Menu adalah bagian dari IDE yang terdiri dari perintah – perintah untuk mengatur IDE, mengembangkan, memelihara dan mengeksekusi program. Di dalam menu, perintah – perintah dikelompokkan kedalam beberapa bagian sesuai jenis perintah tersebut.

b. Toolbar

Toolbar fungsinya sama seperti fungsi dari menu, hanya saja pada toolbar pilihan – pilihan berbentuk icon. Untuk memilih suatu proses yang akan dilakukan, User tinggal mengklik icon yang sesuai dengan proses yang di inginkan.

c. Toolbox

Toolbox adalah tempat dimana kontrol – kontrol dan komponen – komponen diletakkan. Kontrol dan Komponen yang terdapat pada toolbox dipakai dalam pembuatan program aplikasi. Untuk membuat objek kontrol dan komponen pada form program aplikasi diambil dari kontrol – kontrol yang ada pada toolbox.

d. Server Explorer

Server Explorer adalah bagian tempat untuk mengatur hal – hal yang berhubungan dengan server dan database server.

e. Solution Explorer

Solution Explorer memberikan tampilan daftar file – file project yang sedang dibuat sehingga dapat diakses langsung. Pada windows solution explorer terdapat beberapa tombol pada toolbar dan tree yang berisi daftar file –file yang digunakan dalam project.

f. Properties Windows

Properties windows adalah tempat untuk daftar properti setiap objek kontrol dan komponen. Properties window juga dipakai untuk mengatur properti objek kontrol dan komponen yang dipakai. Dengan properties window, User dapat mengubah properti yang nantinya akan dipakai sebagai default objek kontrol dan komponen pada waktu pertama kali program dieksekusi.

g. Form

Form adalah tempat user membuat tampilan (user interface) untuk program aplikasi.

II.9 Pengertian Basis Data

Menurut Kusrini dan Andri Koniyo (2007 : 140) Basis Data adalah suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data dengan cara-cara tertentu sehingga mudah untuk digunakan dan ditampilkan kembali, dapat digunakan untuk satu atau lebih program aplikasi secara optimal, data dapat disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya, serta disimpan sedemikian rupa sehingga penambahan, pengambilan dan modifikasi data dapat dilakukan dengan mudah dan terkontrol.

II.10 SQL Server 2005

Menurut Agus Saputra (2012 : 11) Microsoft SQL Server merupakan salah satu produk RDBMS. RDBMS memiliki kepanjangan *Relational Database Management System*.

Microsoft SQL Server mendukung *SQL* sebagai bahasa pemroses *query*. *SQL* merupakan bahasa standar *international* untuk proses *query database* dan *SQL* ini sudah banyak sekali digunakan pada hampir semua aplikasi, baik itu *e-commerce*, pendidikan, organisasi, pemerintah, atau bahkan personal sekalipun.

Berikut adalah gambar tampilan awal dari *Microsoft SQL server 2005*.



Gambar II.7: Tampilan Awal *Microsoft SQL Server*.

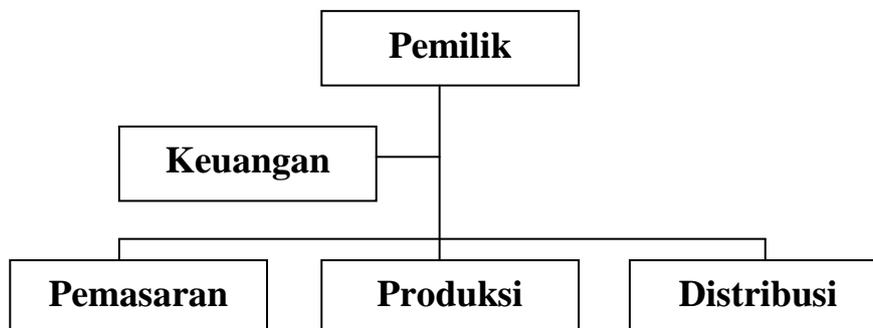
II.11 Sejarah Perusahaan

UD. Anugerah Lestari adalah usaha dari Ibu Rohaida sekeluarga, UD. Anugerah Lestari Mulai usaha pada tahun 1980 yang dikelola sendiri oleh ibu rohaida sekeluarga dengan modal sendiri, usaha ini bergerak di bidang industri pembuatan topi, segala jenis topi di produksi di UD. Anugerah Lestari ini. Pada awal beroperasi usaha ini baru mempekerjakan 1 orang karyawan yang bertugas sebagai staf penjual, kemudian pada tahun 1982 mulai memperkerjakan 1 orang karyawan lagi sampai akhirnya saat ini ibu rohaida sudah mempunyai 4 orang karyawan. Pemasaran produk topi ini hanya mencakup wilayah kota medan saja, meski terbilang usaha kecil menengah tapi industri topi Ibu Rohaida ini sudah berkembang dan banyak mendistribusikan produknya ke pasar-pasar besar di kota Medan. Ibu Rohaida banyak menerima pesanan dengan jumlah yang besar.

Produk topi yang dipasarkan oleh UD. Anugerah Lestari ini terbilang bagus dan mampu bersaing dengan produk-produk topi yang ada di kota medan. UD. Anugerah Lestari ini bukan hanya melayani pembelian yang langsung di lakukan di toko saja, tetapi juga menerima pesanan yang siap di antar langsung ke pada konsumen.

UD. Anugerah Lestari beralamat di Jl. Puskesmas No. 01 Tembung sebagai rumah produksi dan tempat pendistribusiannya sendiri berada di pusat Pasar lantai 1 Medan Mall.

II.12 Struktur Organisasi



Gambar II.8 : Struktur Organisasi UD. Anugerah Lestari.

- **Keuangan**
Bagian yang bertanggung jawab mengurus arus masuk dan keluarnya uang dalam perusahaan.
- **Pemasaran**
Bagian yang bertanggung jawab untuk memasarkan produk, menerima pesanan dan menyampaikan pesanan kepada bagian produksi.
- **Produksi**
Bagian yang bertanggung jawab untuk membuat produk topi serta menghitung kebutuhan bahan yang diperlukan untuk memenuhi permintaan konsumen.
- **Distribusi**
Bagian yang bertanggung jawab untuk menyiapkan bahan baku produk topi agar bagian produksi bisa lancar menghasilkan topi yang berkualitas.